

This Page Is Inserted by IFW Operations  
and is not a part of the Official Record

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning documents *will not* correct images,  
please do not report the images to the  
Image Problem Mailbox.**

(11)特許出願公開番号

(43)公開日 平成6年(1994)8月12日

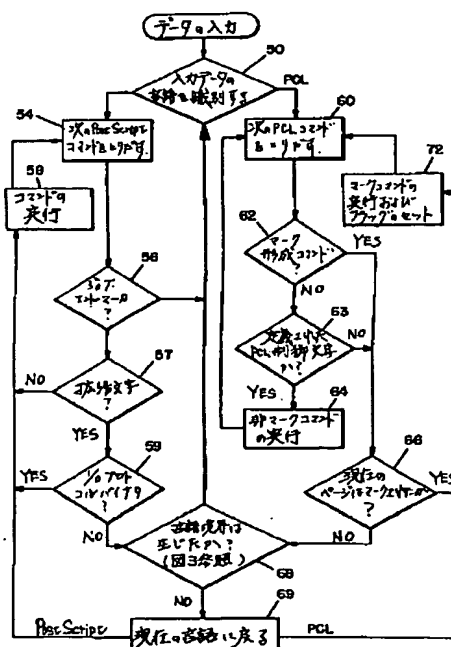
(51)Int.Cl. <sup>5</sup>	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 13/10	3 3 0 B	8133-5B		
3/12	C			
9/45				
13/00	3 5 1 F	7368-5B		
		9292-5B		
			G 0 6 F 9/ 44	3 2 0 A
			審査請求 未請求 請求項の数1	FD (全 9 頁) 最終頁に続く

(33)優先權主張国 米国 (US)

アメリカ国アイダホ州ポイジー・ラムロッド・ロード・ダブリュー 12083

[最終頁に続く](#)

【構成】本発明では、制御言語を認識し、ＰＣＬである場合、入力データ・ストリームからのコマンドよりＰＣＬであるかページにマーキングを行うために呼び出されたものであるか、そしてマーキングが既に施されているかどうか判定する。そうであれば、コマンドと後続のデータについてジョブ境界の発生について調べる。したがって、フォーマット化されたページ・データの開始時にジョブ境界について判定される。ジョブ境界の検出は、入力データ・サンプルを調べ、シンタクスが現在の制御言語が新規のものかを判断することによって行われる。シンタクスが現在のものと異なると、境界が検出されたことになる。



## 【特許請求の範囲】

【請求項1】入力データ・ストリーム中に現れうる複数の制御言語を、受信し、解釈し、応答するように設計された周辺装置が、第1の制御言語を解釈している間に、前記第1の制御言語と第2の制御言語との間の言語境界を識別する方法において、ページ上にマーキングを要求しているかどうかを判定するため、前記データ・ストリーム中のコマンドを解釈し、そうであれば、前記コマンドが前記ページ上にマーキングをおこなったかどうか判断し、前記ページ上にマーキングがなされていない場合、言語境界に遭遇したかどうかを判定し、そうであれば、入力データの制御言語を識別することから成る周辺装置のための制御言語の境界判定方法。

## 【発明の詳細な説明】

## 【0001】

【産業上の利用分野】本発明は、多重制御言語(multiple control languages)をサポートする周辺ユニットに関係し、更に詳細には、異なる制御言語を使用するジョブの間の境界(boundary)を識別するための方法及び装置に関するものである。

## 【0002】

【従来技術とその問題点】現代のデータ処理ネットワークは、種々のタイプのコンピュータを通信ネットワークによって相互接続することを可能にする。例えば、プリンタ、プロッタ等の周辺ユニットは、同様に、通信ネットワークに接続され、接続されたコンピュータ用リソースとして役立つ。これらの周辺装置は、種々のコンピュータから受信したコマンドを理解して、これに対し応答することが可能でなければならない。接続された全てのコンピュータが同じ周辺制御言語を採用する場合には、周辺ユニットは、その言語に対してのみ認識および応答することが必要である。しかし、コンピュータが異なる周辺制御言語を用いる場合、周辺ユニットは、受信する異なる言語に対して認識および応答ができるように設計されていなければならない。

## 【0003】米国特許出願番号07/825,479

(Language Identification System and Method for a Peripheral Unit)では、周辺ユニットが受信するデータ・ブロックの構文(syntactical)および文脈(contextual)の解析によって制御言語を識別できるようにするシステムが開示されている。このシステムは、予期される各言語に対して1つのポーター(voter)モジュールを有し、各ポーター・モジュールは「ディサイダー(decider)」モジュールへ、特定の言語に対して賛成または反対の投票を提供する。ディサイダー・モジュールは、各ポーター・モジュールからの入力の解析の際、周辺ユニットを、ポーター・モジュール入力に従ってディサイダーの制御言語プログラムのうちの1つに対して可能(enable)にするか、或は、識別できる言語がない場合には、周辺ユニットの動作を中止させる。前述の統合論的及び文

脈上の分析システムを確実に作動させるためには、同じでない制御言語を用いるない場合には、周辺ユニットの動作を中止させる。上述の構文及び文脈解析システムを確実に作動させるためには、異なる制御言語を採用するジョブの間の境界が正確に識別されなければならない。

【0004】多重制御言語をサポートする周辺ユニットで現在使用されている1つの方法では、「タイムアウト(期限切れ)」ジョブ分離と呼ばれる方法が用いられている。この手順では、周辺ユニットが次の言語を決定することが可能になる状態に入る以前にジョブの間において、期限切れとなる一定の時間を要求される。次のジョブが期限切れが終了する前に現れた場合、周辺ユニットは言語変化がなく、そして、数百ページの読みにくいテキストを与えるようなミスプリントが生じたと解釈する。

【0005】多重周辺制御言語を用いる多くのシステムにおいては、周辺ユニットに入力されるジョブに対して補助記憶装置を供給するために、通信ネットワークと周辺ユニットとの間にバッファ・デバイスが配置される。このようなバッファ・デバイスは、しばしばタイムアウト・ジョブ分離操作をサポートしない。従って、周辺ユニットは、バッファ・デバイスから順に到着するほぼ又は全くタイムアウト分離のない複数のジョブを受信することもあり得る。

【0006】一般に用いられる2つのプリンタ制御言語は、ヒューレット・パッカード・カンパニーのPCL(Printer Command Language)と、PostScript(米国アドビ社の商標)である。PostScript言語のジョブは一般にジョブ・エンド・マーカ(end-of-job)を使用し、それによって、複数の印刷ジョブの描写を簡素化する。これとは対照的に、PCLのジョブは、ジョブ・エンド・マーカを使用しないことが多い。いずれの場合も、現在のジョブと後続のジョブとの間で制御言語が変化した場合、現在のジョブのエンドが適切にマークされていないと、連続するジョブの制御言語はイネーブルされない。

## 【0007】

【発明の目的】従って、本発明の目的は、制御言語ジョブ境界の発生についてチェックするため、入力データ・ストリームをサンプリングするタイミングを決定するための周辺ユニットのための改良された方法及びシステムを提供することにある。本発明の他の目的は、入力データが現在の制御言語または新規の制御言語のどちらで表現されているかを決定するための改良された言語境界識別手順を提供することにある。本発明の更に他の目的は、ジョブ・エンド・マーカを用いない、周辺制御言語のための改良された言語境界識別手順を供給することにある。

## 【0008】

【発明の概要】周辺ユニットは、入力データ・ストリームの中で、複数の制御言語を受信し、そして、各制御言

語に応答するように機能する。周辺ユニットは、制御言語ジョブ境界の発生をいつチェックするのかを決定する。

【0009】PCLでは、制御言語ジョブ境界の発生をいつチェックするのかを決定するステップを次に示す。入力データ・ストリームから次のコマンドを識別し、識別したコマンドがPCL制御キャラクタあるいはページ上に形成されるマークのための呼び出しであるか判断し、さらに、ページが以前にマークされているかどうかを判断する。前述の判断基準が満たされると、コマンド及び後続のデータを、ジョブの境界が生じるかどうかを決定するために調べる。従って、データのフォーマット化されたPCLページの開始時に、ジョブの境界が生じたかどうかを決定する。

【0010】PostScriptでは、制御言語ジョブ境界の発生をいつチェックするのかを決定するステップは、次のとおりである。データ・ストリームをプリスキャン（予備走査）し、次のコマンドがジョブ・エンド・マーカであれば、PostScriptから出て、入力データの言語を識別するために戻り、また、次のコマンドが拡張文字であって、現在のI/Oプロトコルが2進数でない場合、ジョブ境界が生じたかどうかを決定するために、コマンド及び後続のデータを調べる。

【0011】ジョブ境界は、入力データ・サンプルを調べ、そして、そのシンタックスが現在のジョブの制御言語の特徴であるかまたは異なる制御言語のものであるかを決定する。入力データ・サンプルのシンタックスが現在の制御言語のシンタックスと異なる場合、ジョブ境界が生じている。

【0012】

【発明の実施例】図1では、パーソナルコンピュータ10、12、14は、ローカルエリアネットワーク（LAN）16または他の同様な通信ネットワークによって相互接続されている。プリンタ18は、プリント・ジョブ・バッファ・デバイス（printjob buffer device）19を介して、LAN16に接続されている。プリント・ジョブ・バッファ・デバイスによっては、内部通信機能を有し、これらはLAN16を必要としないことに注意されたい。

【0013】プリンタ18は、パーソナルコンピュータ（PC）10、12、14の夫々にプリントのサービスを提供する。本願明細書では、本発明をプリンタ18に適用する場合について説明するが、接続されたコンピュータに対してサービスを提供する他の周辺ユニットに対しても、本発明を同様に利用できることは明らかである。このような他の周辺ユニットとは、例えば、プロッタ、ファクシミリ・ユニット等がある。

【0014】コンピュータ10、12、14は、それぞれ異なったプリンタ制御言語を用いることが可能である。同様に、これらコンピュータは1台で複数のプリンタ制

御言語を採用することもできる。本発明を説明するために、PC10は、ヒューレット・パッカード・カンパニーのPCLとして知られているプリンタ制御言語を採用し、PC12はPostScriptプリンタ制御言語を採用するものと仮定する。PC14は、他のプリンタ制御言語を採用するものと仮定する。

【0015】PC10、12、14のうちの1つがプリント・ジョブを送信する場合、プリント・ジョブは、LAN16を介して、プリント・ジョブ・バッファ・デバイス19へ送られる。この場合、プリント・ジョブは待ち行列を作り（queued）、プリンタ18へシリアルに供給される。

【0016】プリンタ18は、中央処理装置（CPU）20、入出力（I/O）モジュール22、RAMモジュール24及びROMモジュール25、27を有する。もちろん、ROMモジュール25、27は、単一のROMとして構成されてもよい。これらの各ユニット/モジュールは、バス26を介して、プリント・エンジン（print engine）28まで相互接続される。ROMモジュール25は、言語識別手順のために用いられるプログラムを有し、ROM27は、言語境界識別のためのプログラムを有する。RAM内の30で示される1つの領域は、PC10、12、14のうちの1つから受信するデータのブロックを記憶する領域である。ROM25内のファームウェアは、プリンタ18に対して言語識別手順を具体化し（embodies）、複数のポーター（voter）モジュール34、36、38から構成される。これらはそれぞれ採用が予測されるプリンタの言語の解析および解釈に特別に割り当てられる。ディサイダ（decider）モジュール40は、ポーター・モジュール34、36、38からの出力を受信し、そして、それらのポーター・モジュール入力に基づいて、RAM領域30に記憶されているデータ制御言語のうちのどの制御言語が最も類似していると識別されるかを決定する。次に、ディサイダ・モジュール40は、CPU20に対して、決定がくだされた言語を解釈し、実行することを可能にする構文デコーディング・ソフトウェアへアクセスするように指示する。言語識別手順の動作、多様なポーター・モジュール及びディサイダ・モジュールの動作については、例えば、前述した米国特許出願にも詳細に説明されている。

【0017】言語識別手順を適切に動作させるためには、各ジョブの境界が明確に識別されなければならない。PostScriptジョブは、典型的には、ジョブ・エンド・マーカを使用するので、このような識別は一般に簡単である。対照的に、PCLジョブは、ジョブの終端を識別するためにジョブ・エンド・マーカを採用していないので、別の制御言語（すなわちPostScript）への切り換えは容易ではない。本発明は、PostScriptがそのコマンド（同じくデータ）を具体化するためにASCII文字群を使用している事実を利用するものである。しかし、

PostScriptの文法解析(parser)は、定義されたASCII群はコマンドとして認識され、解釈されるが、PCL文法解析では、ASCII文字を表現された特定の文字に従ってページをマークするためのコマンドとして認識、解釈される。従って、プリンタがPCL文法解析によって制御される場合、コマンドがページ・マークを必要としないコマンドでない限り、プリンタはページをマークすることにより、PostScriptのコマンドの最初の文字に応答する。加えて、本発明は、PCLが多くのコマンドを開始するために拡張文字(ASCII 27)を用いること及び非2進数PostScriptデータ・ストリーム中に拡張文字が存在することは極めてまれであるということも利用する。従って、PCLジョブを開始するには感知された拡張文字を使用することが極めて好ましく、この場合には、PostScriptジョブが適切なジョブ・エンド・マーカで終結しなかったことにより、PostScriptジョブが「不良形成された(mal-formed)」ものと仮定される。

【0018】ROM27における言語境界識別手順は、

(1) PCLと、異なる制御言語を用いる後続のジョブとの間のジョブ境界の識別を可能とし、(2) PostScriptと、異なる制御言語を用いる後続のジョブとの間のジョブ境界の識別を可能にする。言語境界識別手順(CPU20との組み合わせ)によって用いられるステップを図2に示す。言語境界(LB)文法解析42はROM27内に含まれており、そして、境界識別手順において支援する(以下に詳細に説明する)。

【0019】最初に、図2において、プリンタ18(図1)に示される言語識別手順により、現在処理中の入力データ・ストリームの言語を識別する(デシジョン・ボックス50)。識別された言語がPostScriptであれば、図2の左側に示す手順を下方に向かって繰り返す。これとは対照的に、現在識別された言語がPCLであれば、システムは図2の右側に示す手順を下方に向かって繰り返す。識別された言語がPostScriptである場合、手順は次のコマンド(ボックス54)にアクセスする。非2進数(non-binary)PostScriptジョブには、拡張文字が存在することは極めて稀である。次のコマンドがジョブ・エンド・マーカ(ボックス56)を指示するならば、現在のジョブを終結し、手順サイクルはボックス50に戻り、新たにジョブの言語を識別する。コマンドがジョブ・エンド・マーカでないならば、拡張文字(ASCII 27)であるかどうかを見るために、次のコマンドを試験する(デシジョン・ボックス57)。コマンドが拡張文字で、しかも、現在の1/0プロトコルが2進法でないならば(ボックス59)、制御言語境界が生じたかどうかを決定するためにデシジョン・ボックス68を呼び出す。

【0020】現在識別済みの言語がPCLであるとデシジョン・ボックス50によって示された場合、手順は次

のコマンドを取り出すように繰り返される(ボックス60)。ここで、PCLは文字をプリントするためのコマンドとして、ASCII表現された文字が認識する。従って、ページ上に1つのマークがプリントされるべきであることを示すPCLコマンドのグループの1つであるかどうかを決定するために、取り出したコマンドを構文的に解析する(デシジョン・ボックス62)。そうでない場合、ページをマークしないが、言語は継続してPCLであることを高い信頼性をもって指示する1組の予め定義されたPCL制御文字と当該コマンドが一致するかどうかを決定するために、当該コマンドを更に解析する。それらのコマンドは、ASCII 110進数の値は0、7-15、27(即ち、「拡張」文字)を有している。このようなコマンドを見つけた場合(デシジョン・ボックス63)、決定されたPCLのマーキングしないコマンドが実行され(ボックス64)、そして、次のコマンドを取り出すために手順が繰り返される。

【0021】ページのマーキングを要求するコマンド(例えば、ASCII文字)を認識し(デシジョン・ボックス62)またはデシジョン・ボックス63に示された解析手順によって、当該コマンドが定義されたPCL制御文字でないと示す場合、現在のページが既にマーキング済みかどうかを決定するため繰り返す(デシジョン・ボックス66)。そうでないならば、可能なジョブ境界に到達したものと仮定する。マークがページ上に配置されると、直ちに、マーキング・フラグをセットし、それ以後、ページがマークされたかどうかを決定するためにはマーキング・フラグがセットされたかどうかを調べる。現在のページが完成し、プリントされるまで、このフラグはセットされたままに維持され、プリントされた時点でマーキング・フラグをリセットする。

【0022】デシジョン・ボックス68では、コマンド及びシンタックスが「現在の」制御言語または「新規な」制御言語における「キー」を表すかどうかを決定するためにデータ・サンプルを解析することにより、ジョブ境界が生じたかどうかを決定する。「キー」は、制御言語のコマンド・ルール内で定義された単一の文字または文字群である。ボックス68では、データ・サンプル中からキーの識別を継続しておこない、そして、データ・サンプルの終端が生じたかまたは予め決められた現在の言語または新規な言語のキー・ウエイトしきい値(key-weight threshold)を越えるまで、現在の言語及び新規な言語のウエイトを蓄積する。現在の言語または新規な言語のキーウエイトしきい値を越える前に、データ・サンプルの終端が生じていたら、ジョブ境界は生成されていない。現在の言語のキーウエイトしきい値を越えると、ジョブ境界は生成されていない。新規な言語のキーウエイトしきい値を越えると、ジョブ境界が生成されたことが指示された場合、入力データの言語を識別す

るためにボックス50が呼び出されて、新規な言語がイネーブルとなり、そして、イネーブルの言語を用いる新規なジョブが開始される。

【0023】上述より、ジョブ境界に関してデータ・ストリーム中の一点を探しだす機構が可能となり、そして、その点において、制御言語が現在の制御言語と同じであるか、または、新規な制御言語と同じであるかを決定するために、データの簡単な解析がおこなわれることになる。

【0024】図3のフローチャートでは、図2のデシジョン・ボックス68に示されたソフトウェアによって行われる動作を示す。以下に手順に際して識別される機能は、ROM27内のLB文法解析(LB parser)42によって実施される(図1参照)。LB文法解析42は、入力データ・ストリームの開始時に、データ・サンプルからデータを読み取る。LB文法解析42は一連のキーを認識し、データ・サンプルを解析すると各キーをカウントする。キーが生成されると、アキュムレータにインクリメントが加算される。各キーを受信し、対応するアキュムレータがインクリメントされた後、データ・サンプルがPostScriptまたはPCLであるかどうかを決定するために、アキュムレータ値を手順にしたがって評価する。LB文法解析42は、PostScriptまたはPCLによって明確な決定ができるまで、入力データを文法解析する。明確な決定がくだすことが可能となると、それ以上の解析は完了する。

【0025】特定のキーにはそれぞれウェイト値(weight value)が割り当てられ、それが検出されると、キーのタイプに割り当てられたウェイトの量だけそれぞれ対応するアキュムレータがインクリメントされる。従って、あるタイプのキーは他のタイプのキーよりもアキュムレータ値に及ぼす影響が大きく、前者は後者よりも言語の存在に対してより指示される。LB文法解析42によって認識され、PCLとPostScriptとを区別するために用いられるキーを次に示す。キーは、PCLディスクリミネータ(discriminator)及びPostScriptディスクリミネータとに分類される。特に注記しない限り、キーの生成毎にキー・ウェイトがカウントされる。

【0026】PostScriptキー

(1)特殊ジョブ開始制御文字

次の制御文字シーケンスのうちの1つが発生すると、データ・サンプルの開始時に一度だけカウントされる。:

^A、^C、^D、^S、^O、^T

(2)PostScriptヘッダコメント

サブストリング「%!PS」と同じラインのテキスト・ストリング又はサブストリング「%!PS」、「%!」、「EPSF」。

(3)非同期制御文字(asynchronous Control Characters)は、データ・サンプルの残りを評価する前に、カウントされ、ストリップされる。:^T、^C

(4)PostScript一意キーワード

予め定義されたPostScriptオペレータは、一般に、PostScript言語の外側では発見されない。

(5)経験的なPostScriptキーワード

PostScriptアプリケーションは使用するが、PostScript言語の一部ではない周知の非英語(non-English)ワード。

(6)PostScript文書化キーワード

AdobeによってPostScript文書化キーワードとして定義されたワード。これらのワードは、PostScriptコメントにおける最初のワードとして発生する場合に限り、カウントされる。

(7)PostScriptコメント

テキストは「%」で始まり、ラインの終端を通過して継続する。「フォームフィード拡張」文字に遭遇した場合には、コメントの評価は早期に完了される。

(8)PostScript中立キーワード

PostScript言語以外の意味を有し、通常の英語に含まれることもある、予め定義されたPostScriptオペレータ。

【0027】PCLキー

(1)拡張文字

PCL拡張シーケンス・コマンドの開始時に、用いられるASCII値が27の文字。

(2)フォームフィード文字

現在フォーマット化されたPCLページをプリントさせるASCII値が12の文字。

(3)未知ワード

予め定義されたPostScriptキーとマッチしないテキスト・ストリング。

【0028】読み取られた最初の文字が特殊制御文字であった場合には、上記の特殊ジョブ開始制御文字は、PostScriptジョブの開始を強く指示する。従って、このキーはLB文法解析42によってただ1度だけカウントされるが、データ・フローの最初にみつけた場合には大きく加重される。これとは対照的に、中立キーワードはPostScriptばかりでなく他の言語も同様に指示することができるので、わずかに加重されるだけである。しきい値は経験に基づいて決定することも可能である。これにより、蓄積されたキーワード値ウェイトの合計が限界を越えた場合、PostScript言語が決定されていると判定する。LB文法解析42は、解析手順を短縮法(foreshorten)で表わし、一時的な選択以外の言語であることを明確に示す特定の文字に遭遇した場合には、一時的な言語の選択を拒否することができる。LB文法解析42によって実施される解析には、言語識別にとって同様に重要であるキーの文脈上の関係が考慮されていない。従って、上述の特許出願に開示される詳細な言語識別手順によって、誤りが実施されたことが指示された場合には、判定が逆転することもあり得る。それにも拘わらず、上述の分析は、ジョブ言語境界の可能性が高いことを示

すには十分である。

【0029】図3に示す前述の手順において、ボックス80はデータ・サンプルが現在の制御言語または別の制御言語を表すかどうかを決定するためにLB文法解析42によって実施されるキー識別手順を示す。このキー識別手順は、言語決定が可能になるかあるいはデータ・サンプルの終端に到達する時まで、データ・サンプル内において認識されたキーに関するキーウエイトを蓄積することによって実施される。データ・サンプルの終端に到達し、しかも、言語決定がおこなわれなかった場合には、ジョブ境界は生成されない。

【0030】この手順は、データ・サンプルに対して次のように繰り返される。データ・サンプル中の次の識別されたコマンド(ボックス80)が現在の言語キーにマッチする場合(デシジョン・ボックス82)、そのキーへのキーウエイトは、現在の言語総キーウエイトに蓄積される(ボックス88)。データ・サンプル中の次の識別されたコマンドが、新規な言語キーとマッチする場合には(デシジョン・ボックス82)、キーウエイトは、新規な言語総キーウエイトに蓄積される(ボックス84)。現在の言語キーウエイト総量が予め決められたしきい値に等しいか、あるいは、それより大きい場合には(デシジョン・ボックス90)、ジョブ境界はなく(ボックス92)、現在の制御言語に対して処理を継続する。新規言語のキーウエイト総量が、予め決められたしきい値に等しいか、或いは、それより大きい場合には(デシジョン・ボックス86)、ジョブ境界が生じ(ボックス87)、図2のボックス50の手順をでて、入力データ・サンプルの新規な制御言語の識別をおこなう。

【0031】以上の説明は本願発明の例を示すに過ぎないことを理解されたい。当業者であれば、本発明に趣旨から逸脱することなく、さまざまな変更および改善が可能であることは自明のことである。

#### 【0032】

【発明の効果】以上説明したように、ジョブ・エンド・マーカを用いない、周辺制御言語のジョブ境界を識別し、周辺装置のための入力データ・ストリームをサンプリングするタイミングを決定することを可能とする。さらに、本発明によって、入力データが現在の制御言語または新規の制御言語のどちらで表現されているかを決定することを可能とする。

#### 【図面の簡単な説明】

【図1】本発明の周辺装置のための制御言語の境界判定方法をおこなうシステムの概略図。

【図2】第1図のシステムの制御言語の境界判定の動作を説明するフローチャート。

【図3】第2図に加えて、文法解析の動作を説明するフローチャート。

#### 【符号の説明】

10、12、14:PC

16:LAN

19:プリント・ジョブ・バッファ

20:CPU

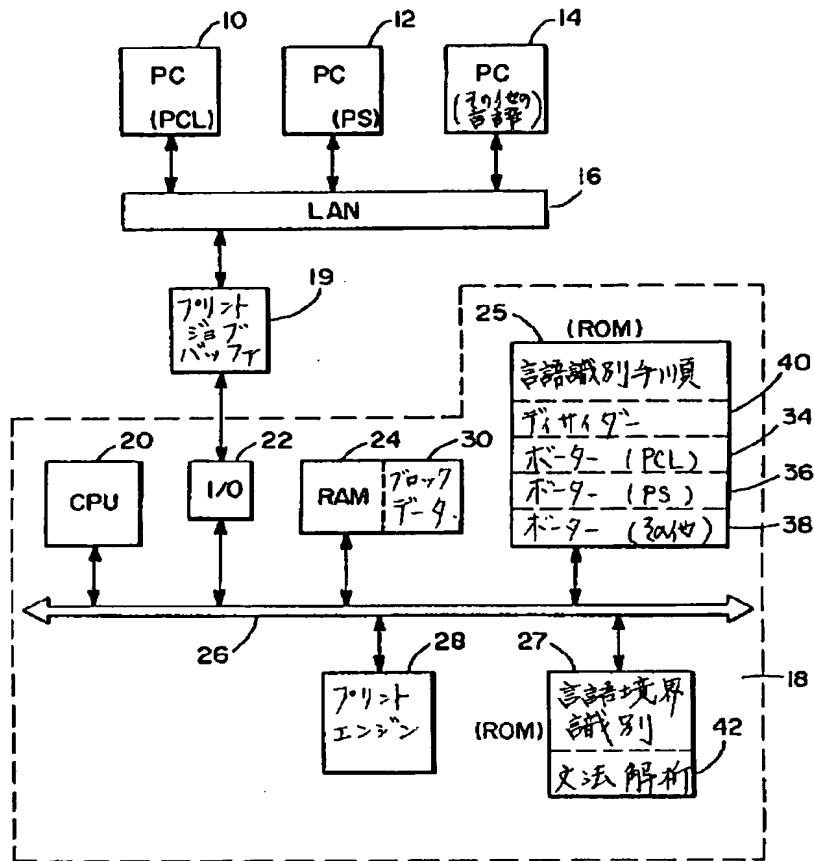
22:入出力装置

24:RAM

25、27:ROM

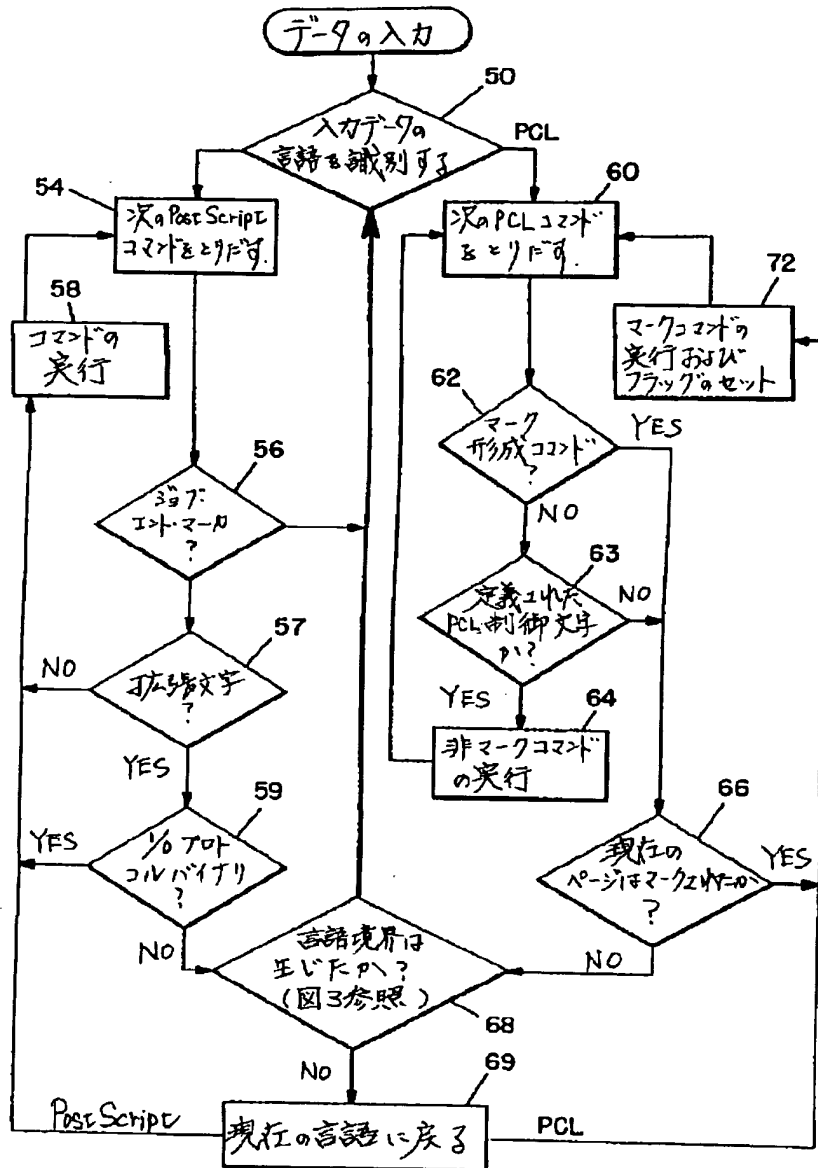
28:プリント・エンジン

【図1】

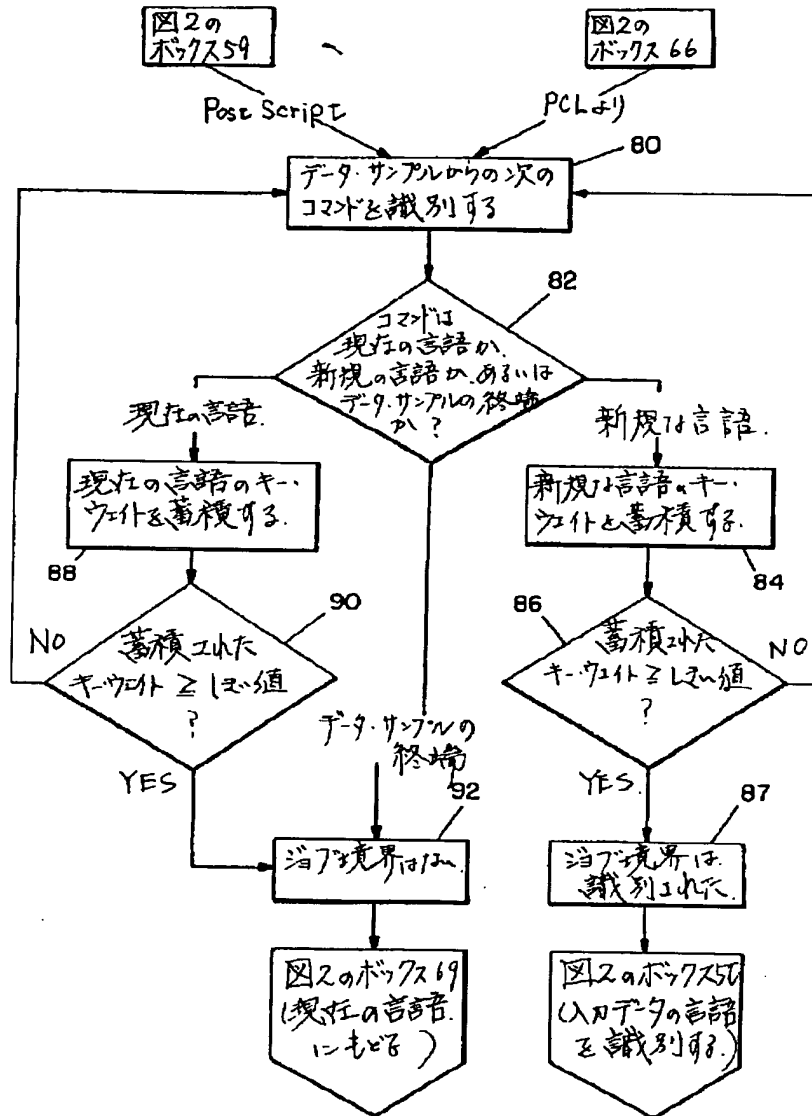




【図2】



【図3】



フロントページの続き

(51) Int. Cl. 5

G 0 6 F 15/00

識別記号 庁内整理番号

3 1 0 T 7459-5L

F I

技術表示箇所

(72) 発明者 マイケル・イー・スローン  
 アメリカ国アイダホ州ボイジー・クリー  
 ウェイ 5026